Omnixx™ Interface Specification

Datamaxx Standard Embedded Object® DSEO-2020®
Version 1.0

Datamaxx® APPLIED TECHNOLOGIES

Leading Law Enforcement Technology

www.Datamaxx.com

Published by:

Datamaxx Applied Technologies, Inc.
3780 Peddie Drive
Tallahassee, Florida, 32303
(850) 575-1023
www.datamaxx.com

Revision Levels:

Revision 0, October 1999

Revision 1, September 2000
Section 2.3.1: corrected error in example

Table of Contents

1 Introduction

The purpose of this document is to define a standard that can be implemented to support the exchange of non-text objects such as mug shots, pictures of stolen property and fingerprints in the Law Enforcement environment. Historically, Law Enforcement message switching has been based on standards and formats defined by the Federal Bureau of Investigation's National Crime Information Center (NCIC) and the National Law Enforcement Telecommunications System (NLETS). Until recently, these systems were text based and couldn't process messages containing such embedded non-text objects since it was possible, and quite likely, that they would contain control characters that would interfere with the bisynch line protocol. The user terminals in most states had similar problems with line protocols such as 8A1/8S1, Uniscope, 3270, poll-select and others. In addition, few if any of the terminals were able to display images. Most Control Terminal Agency (CTA) switches were not designed to process messages containing embedded non-text objects and would probably reject any if received. One of the main reasons for this is there weren't any standards for including non-text objects in Law Enforcement messages.

With NCIC 2000, the NLETS upgrade and the widespread use of network protocols such as TCP/IP and LU 6.2, some of these barriers have been removed. NCIC and NLETS now have standards for including non-text objects in messages exchanged with CTAs. And since TCP/IP and LU 6.2 are "data transparent", they can transport any data since they are not susceptible to interference from binary data in the payload. Unfortunately these standards, while adequate for interfacing with NCIC and NLETS individually, do not provide a comprehensive approach for CTAs to use within their own system – that is, in-state traffic between the CTA switch and its end users. This is due to the following issues:

a) The NCIC and NLETS standards for embedding photographs and fingerprints in messages are based on completely different approaches and, in the opinion of Datamaxx, the NLETS standard is much too complex for in-state use.

b) The NCIC standard does not include a provision for removing images from messages copied or sent to devices that can't process them.

c) CTAs may want to allow objects other than those supported by NCIC and NLETS to be supported within their own state.

d) CTAs in different states will likely need to contend with additional standards for exchanging driver's license photos with their Department of Motor Vehicles (DMV), mug-shots and fingerprints with their Computerized Criminal History (CCH) systems and so on. As other data sources adopt still other standards, all client devices (workstations, MDCs, CAD systems, remote hosts and so on) would need to be updated to recognize the new formats. It is quite likely that not recognizing and properly processing a message containing a non-text object will cause fatal problems for an application.

e) CTAs need a standard that allows their users to exchange messages such as in-state AMs that contain these objects.

The remainder of Section 1 provides an example of the problems these issues could cause and the objectives of the Datamaxx Standard Embedded Object (DSEO-2020®) as a solution.

Section 2 contains specifications for the DSEO-2020® object. It also revisits the example provided in Section 1 and shows how DSEO-2020® eliminates the problems.

Datamaxx and Computer Projects, Inc. (CPI), our partner in Law Enforcement solutions, both support this standard in our products and advocate its use for all in-state communications. We recommend CTAs adopt it wherever possible as their standard for exchanging all embedded objects, including images, mugshots, driver's license photos and fingerprints, between the message switch and workstations, remote

hosts and in-state databases, in both directions. Datamaxx will also support similar standards that CTAs have adopted as long as they adequately address the objectives described in Section 1.2

1.1 Example of Problems Caused by Multiple Standards

This section illustrates the problems associated with using multiple standards for embedding non-text objects in messages exchanged between a message switch and its clients. It shows using the NCIC standard along with a standard currently used in one state for retrieving operator license photos. The latter standard will be referred to as AS (another standard). In both cases, the image is exchanged in JPG format.

The NCIC standard for identifying images in a response are well documented in the *NCIC 2000 Message Book*. In summary, images are identified by a line beginning with "IMR/". This is followed immediately by: 1 byte IMT, 47 byte upper top text (UTT), 47 byte upper bottom text (UBT), 47 byte lower top text (LTT), 5 byte ASCII encoded decimal image size indicator and a JPG image. For example (note that the second line feed after the ORI is not shown to simplify the example):

```
1L01TESTX
AS1234567
MKE/IMAGE
IMR/MNAM:R0BERTS, RICARD0               DOB:19
710202RAC:W HGT:511     WGT:195     DOI:199903
29     NIC:W150005877 IMN:I125000155
        02410<2410 byte JPG image>
```

Images embedded using the AS standard are identified by: the presence of the pattern "IMR/" anywhere in the message followed by a 4 byte ASCII encoded decimal image length indicator and a JPG image of that length. For example:

```
DR.ASDMV0000.AS1234567.TXT
NAM: R0BERTS, RICARD0
DOB: 19710202 RACE:W
HGT: 511 WGT:195
IMR/4567<4567 byte JPG image>
```

States typically deliver responses from NCIC and other databases to the originator of the request after adding a delivery header that identifies, among other information, the device that sent the response, such as NCIC or DMV. In this example, the NCIC and DMV responses, respectively, would be sent to ORI AS1234567 as follows:

```
MSG 24680 NCIC      AS1234567 11:00:01 11/01/1999
1L01TESTX
AS1234567
MKE/IMAGE
IMR/MNAM:R0BERTS, RICARD0               DOB:19710
202RAC:W HGT:511     WGT:195     DOI:19990329
 NIC:W150005877 IMN:I125000155                  0
2410<2410 byte JPG image>
```

and

```
MSG 24681 DMV       AS1234567 11:00:02 11/01/1999
DR.ASDMV0000.AS1234567.TXT
NAM: R0BERTS, RICARD0
DOB: 19710202 RACE:W
```

```
HGT: 511 WGT:195
IMR/4567<4567 byte JPG image>
```

ORI AS1234567 would need to scan the message for images based on the source device identified in the delivery header. This would be adequate for the device receiving the response directly (through the switch) from NCIC or DMV. The problems arise if this message is forwarded to another workstation. For example, in many states, a response can be forwarded to another device by prefixing it with an AM header. If both of these messages were forwarded in this manner to ORI AS7654321, they would be received as follows:

```
MSG 13570 AS1234567 AS7654321 11:00:03 11/01/1999
MSG 24680 NCIC      AS1234567 11:00:01 11/01/1999
1L01TESTX
AS1234567
MKE/IMAGE
IMR/MNAM:R0BERTS, RICARD0               DOB:19710
202RAC:W HGT:511     WGT:195     DOI:19990329
 NIC:W150005877 IMN:I125000155                  0
2410<2410 byte JPG image>
```

and

```
MSG 13571 AS1234567 AS7654321 11:00:04 11/01/1999
MSG 24681 DMV       AS1234567 11:00:02 11/01/1999
DR.ASDMV0000.AS1234567.TXT
NAM: R0BERTS, RICARD0
DOB: 19710202 RACE:W
HGT: 511 WGT:195
IMR/4567<4567 byte JPG image>
```

Here's where the problem arises: based on the delivery header, ORI AS7654321 only knows that the message was received from ORI AS1234567. How does it know which set of rules to use to scan for images? What if it used the DMV rules but the response was from NCIC? The message does contain "IMR/" but the images are embedded according to the NCIC standard. The different applications in the client devices such as workstations, CAD systems and remote hosts would each react to this situation differently and there is no guarantee it wouldn't be fatal.

This is but a single example of using the NCIC standard along with a state-unique standard. Even with these two standards, there are many other cases to consider such as how does the switch distinguish between inputs using the NCIC EIM approach as opposed to a forwarded NCIC or DMV response. The complexities will grow exponentially with each additional standard. And each device will need to be designed to deal with these complexities individually.

The Section 2 of this document specifies the Datamaxx Standard Embedded Object (DSEO-2020®) and revisits this example to show how DSEO-2020® avoids these problems.

1.2 DSEO-2020® Objectives

The overall objective of the Datamaxx Standard Embedded Object (DSEO-2020®) is to provide a robust approach for exchanging messages containing embedded non-text objects in a typical Control Terminal Agency (CTA) environment. The approach needs to address the issues described in the Introduction to this standard and avoid the problems illustrated in Section 1.1. Specifically, the approach must meet the following objectives:

a) **NCIC 2000 and NLETS Upgrade Compatibility.** The approach must be compatible with NCIC 2000 and the NLETS upgrade. Compatibility, however, doesn't imply that the standard must include objects embedded in native NCIC 2000 and NLETS formats. The only requirement is that it allows messages to be exchanged with these interfaces according to their standards, even if it requires reformatting between their standards and a CTA's standards.

b) **Standard Object Compatibility.** The approach must be compatible with standard non-text objects such as Microsoft Word (DOC), Visio drawing (VSD), bitmap (BMP), etc. files.

c) **Extensible.** The approach must be easily extended to include new types of objects and the impact of adding them to client applications must be minimal.

d) **Unlimited Number of Objects in a Message.** The approach must allow any number of objects to be embedded in a single message, limited only by the maximum size of the message.

e) **Processing Platform Independent.** The approach must able to be implemented in any processing environment and in any suitable programming language and not depend on specific data representations of the platform.

f) **Communication Protocol.** The approach must be independent of any the communication protocol. The only requirement on the communications protocol is that it be data transparent – not susceptible to interference from any possible character or sequence of characters within the data payload.

g) **Unlimited forwarding of Messages.** The approach must allow messages with embedded objects to be forwarded any number of times.

h) **Simple.** The approach must be as simple as possible while meeting these objectives.

2 DSEO-2020® Specifications

The following sections define the format and processing specifications for the DSEO-2020® object.

2.1 DSEO-2020® Format Specifications

The following table shows how Datamaxx Standard Embedded Objects (DSEO-2020®) are structured. Any number of DSEO-2020® objects may be embedded in a message.

```
[START][OBJECT LENGTH][OBJECT TYPE][DATA LENGTH][DATA][TEXT
LENGTH][TEXT][STOP]
```

Field	Length	Contents
START	4	a) Start pattern b) Always "<DLE>dbo" (hex representation: 0x1064626F)
OBJECT LENGTH	4	a) Length of object from START to STOP, inclusive b) 4 byte unsigned integer in Network Byte Order
OBJECT TYPE	4	a) Type of object contained in the DATA field – see detailed explanation below b) 4 alphanumeric characters; blank-fill fourth character if necessary
DATA LENGTH	4	a) Length of DATA field b) 4 byte unsigned integer in Network Byte Order
DATA	Variable	a) Variable length field containing the embedded object
TEXT LENGTH	4	a) Length of TEXT field b) Must be zero if text is not included c) 4 byte unsigned integer in Network Byte Order
TEXT	Variable	a) Optional, variable length field containing the text associated with the embedded object
STOP	4	a) Stop pattern b) Always "obd<DLE>" (hex representation: 0x6F626410)

2.1.1 Object Types

The DATA field can contain standard or non-standard objects. Standard objects are files that are typically associated with applications running in the Microsoft environment such as Microsoft Word (DOC), Visio drawing (VSD), bitmap (BMP), etc. files. For standard objects, the OBJECT TYPE field contains the extension associated with the file. For example, for Microsoft Word files the OBJECT TYPE is "DOC" and for a BMP file the OBJECT TYPE is "BMP".

Non-standard objects can be defined to fit any need as long as there is agreement among the interfacing parties. At a minimum, the following non-standard types should be supported for NCIC and NLETS messages. See the *NCIC-2000 Message Book* for detailed specifications of NCIC message formats and the *NLETS User's Guide* for specifications of NLETS message formats.

a) **"IMG": non-fingerprint images sent to NCIC.** The DATA field for this OBJECT TYPE includes the 1 character image type designator, 5 character image size indicator and the JPG image data. It contains exactly the same data as will follow "IMG/" in an EIM transaction for entering a non-fingerprint image.

b) **"FIM": fingerprint data sent to NCIC.** The DATA field for this OBJECT TYPE includes the "F", "M" and "V" indicators and associated byte counts, and image data. It contains exactly the same data as will follow "FIM/" in an EIM transaction for entering a fingerprint image.

c) **"IMR": images included in responses from NCIC.** The DATA field for this OBJECT TYPE includes the 1 character image type designator (IMT); 47 character upper top text (UTT), upper bottom text (UBT) and lower top text (LTT) fields; five character image size indicator and the JPG image data. It contains exactly the same data as will follow "IMR/" in a response from NCIC that includes an image.

d) **"NRAP": Standardized NLETS RAP sheets.** The specification for this OBJECT TYPE will be defined once NLETS finalizes its definition of standardized RAP sheets.

2.1.2 Text Field

The TEXT field can be used to associate text with a DSEO-2020® object. For example, when a state's DMV provides an operator's license photo, it would be useful to include the operator's name or other identifying information within the object such that the information is always associated with it.

2.2 DSEO-2020® Processing Specifications

Any system (client or server) that uses Datamaxx Standard Embedded Objects (DSEO-2020®) should process all inputs that could contain the object in the manner described below which addresses how to detect the object and verify its integrity.

1) Scan the message for the START pattern.
2) If the START pattern is detected, determine OBJECT LENGTH from the next 4 bytes.
3) Read the number of bytes specified by OBJECT LENGTH less 8 (the remaining number of bytes in the object after START and OBJECT LENGTH).
4) Perform the following integrity checks. If any fail, it should be assumed that the detected pattern does not indicate the beginning of a DSEO-2020® object and scanning should resume with the bytes immediately following the detected pattern.
   a) The last 4 bytes should contain the STOP pattern.
   b) OBJECT LENGTH, DATA LENGTH and TEXT LENGTH should all be positive integers.
   c) The following should be true:

      DATA LENGTH + TEXT LENGTH = OBJECT LENGTH – 24

   d) OBJECT TYPE should contain alphanumeric or blank spaces only.
5) Process the object per the specific requirements of the application and continue with Step 1 to search for additional objects until no more are found.

Further processing of the object depends on the specific application and could include such actions as displaying the object, storing the object in a file, removing the object if the intended destination doesn't support images, converting the object to another form of representation, etc. For example, if a message switch receives an NCIC enter image (EIM) transaction from a workstation, the IMG field should contain a JPG image formatted as a DSEO-2020® object with TYPE="IMG" and TEXT LENGTH=0. In order to comply with NCIC standards, the switch would reformat the IMG field to NCIC standards before sending it to NCIC.

2.3 Examples using DSEO-2020®

The following examples illustrate use of the Datamaxx Standard Embedded Object (DSEO-2020®) for embedding objects in messages exchanged between a message switch and its clients. In the examples, characters shown in bold are hexadecimal equivalents of their respective byte values (2 hexadecimal characters define a single byte).

2.3.1 Previous Example Revisited

This example revisits the situation described in Section 1.1 illustrating the problems associated with using multiple standards for embedding non-text objects in messages exchanged between a message switch and its clients. It shows the same responses from NCIC and DMV that are delivered to a device in the DSEO-2020® format and the subsequent forwarding of them to another device.

The NCIC response is still formatted to NCIC standards:

```
1L01TESTX
AS1234567
MKE/IMAGE
IMR/MNAM:R0BERTS, RICARD0               DOB:19
710202RAC:W HGT:511     WGT:195     DOI:199903
29     NIC:W150005877 IMN:I125000155
        02410<2410 byte JPG image>
```

The DMV response is still formatted to the hypothetical AS standard:

```
DR.ASDMV0000.AS1234567.TXT
NAM: R0BERTS, RICARD0
DOB: 19710202 RAC: W
HGT: 511      WGT: 195
IMR/4567<4567 byte JPG image>
```

Both responses are delivered to the destination with delivery headers. However, the images and related text have been reformatted per the DSEO-2020® standard. For the NCIC response, the image and all related information (IMT, UTT, UBT and LTT) are encapsulated into an “IMR” object.

The NCIC response would be sent to ORI AS1234567 as follows:

```
MSG 24680 NCIC      AS1234567 11:00:01 11/01/1999
1L01TESTX
AS1234567
MKE/IMAGE
IMR/1064626F00000A10IMR 000009F8MNAM:R0BERTS, RIC
ARD0               DOB:19710202RAC:W HGT:511
WGT:195     DOI:19990329     NIC:W150005877 IMN:I
125000155                  02410<2410 byte JPG im
age>000000006F626410
```

The DMV response would be sent to ORI AS1234567 as follows:

```
MSG 24681 DMV       AS1234567 11:00:02 11/01/1999
DR.ASDMV0000.AS1234567.TXT
NAM: R0BERTS, RICARD0
DOB: 19710202 RAC: W
HGT: 511      WGT: 195
IMR/1064626F000011EFJPG 000011D7<4567 byte JPG im
age>000000006F626410
```

ORI AS1234567 would scan the message for embedded objects based on the DSEO-2020® standard, not the source device. It would scan every message for the start pattern **1064626F**. When detected, the presence and integrity of a DSEO-2020® object could be confirmed by the rules described in Section 2.2. By using DSEO-2020®, the client software has benefited since it only needs to be able to process a single type of embedded object instead of two. The real benefit is realized when the messages are forwarded to another device. If both of these messages are forwarded to ORI AS7654321, they would be received as follows:

```
MSG 13570 AS1234567 AS7654321 11:00:03 11/01/1999
MSG 24680 NCIC      AS1234567 11:00:01 11/01/1999
1L01TESTX
AS1234567
MKE/IMAGE
IMR/1064626F00000A10IMR 000009F8MNAM:R0BERTS, RIC
ARD0               DOB:19710202RAC:W HGT:511
WGT:195     DOI:19990329     NIC:W150005877 IMN:I
125000155                  02410<2410 byte JPG im
age>000000006F626410
```

and

```
MSG 13571 AS1234567 AS7654321 11:00:04 11/01/1999
MSG 24681 DMV       AS1234567 11:00:02 11/01/1999
DR.ASDMV0000.AS1234567.TXT
NAM: R0BERTS, RICARD0
DOB: 19710202 RAC: W
HGT: 511      WGT: 195
IMR/1064626F000011EFJPG 000011D7<4567 byte JPG im
age>000000006F626410
```

When they are received by the second device (AS7654321), they can be processed in exactly the same way they were processed by the first device (ORI AS1234567) since processing is based on the DSEO-2020® standard instead of the original source. There is no confusion or need to know the original source. And the message could be forwarded any number of times without a problem.

The remaining examples illustrate how DSEO-2020® would be used in other situations.

2.3.2 Example: NCIC Enter Image (EIM) Transaction

Workstation AS1234567 sends the switch an NCIC enter image (EIM) transaction from a workstation containing an NCIC IMG image embedded as a DSEO-2020® object. The IMG field contains the IMT indicator, image length, and a 4500 byte JPG mugshot.

The message as sent by AS1234567:

```
EIM.AS1234567.NIC/W987654321.IMT/M.DOI/19991015.I
MG/1064626F000011B2IMG 0000119AM04500<JPG mug hot
>000000006F626410
```

The EIM transaction as it should be sent to NCIC with the DSEO-2020® object reformatted to NCIC standards.

```
1B01987654321012345.EIM.AS1234567.NIC/W987654321.
IMT/M.DOI/19991015.IMG/M04500<JPG mugshot>
```

2.3.3 Example: AM Message

Workstation AS1234567 sends workstations AS7654321 and AS7654322 an AM message via the switch. The message contains a 4K byte JPG photo and 8K byte BMP drawing, both embedded as DSEO-2020®

objects with information entered in both TEXT fields. AS7654321 can receive messages containing embedded objects but AS7654322 can't.

The message as sent by AS1234567:

```
AM.NB1234567.NB2468024.NB1357913.TXT
BE ON LOOKOUT FOR SUSPECT … PER FOLLOWING PHOTO
1064626F00000FC4JPG 00000FA0<4000 byte JPG mug ho
t>0000000CROBERT SMITH6F626410
FOLLOWING SMT ON RIGHT FOREARM
1064626F00001F65BMP 00001F40<8000 byte BMP drawin
g >0000000DRIGHT FOREARM6F626410
```

The message as it should be sent to AS7654321 containing DSEO-2020® objects:

```
MSG 13570 AS1234567 AS7654321 11:00:03 11/01/1999
AM.NB1234567.NB2468024.NB1357913.TXT
BE ON LOOKOUT FOR SUSPECT … PER FOLLOWING PHOTO
1064626F00000FC4JPG 00000FA0<4000 byte JPG mug ho
t>0000000CROBERT SMITH6F626410
FOLLOWING SMT ON RIGHT FOREARM
1064626F00001F65BMP 00001F40<8000 byte BMP drawin
g >0000000DRIGHT FOREARM6F626410
```

The message as it should be sent to AS7654322 with the JPG photo and BMP drawing replaced by text noting their removal:

```
MSG 13570 AS1234567 AS7654322 11:00:03 11/01/1999
AM.NB1234567.NB2468024.NB1357913.TXT
BE ON LOOKOUT FOR SUSPECT … PER FOLLOWING PHOTO
*** IMAGE REMOVED ***
FOLLOWING SMT ON RIGHT FOREARM
*** IMAGE REMOVED ***
```